

# File Aggregation in Enstore (Small Files) Status report 2

<https://cdcvs.fnal.gov/redmine/projects/show/fileaggregation>

**Alexander Moibenko**

# Problem

- Writing or reading a tape mark (EOF) at the end of a file takes about 3 seconds. Writing or reading a full tape of continuous data takes just under 2 hours at top speed.
  - Thus, a tape full of 360 MB files would take twice as long —4 hours.
  - So files ought to be much larger; a few GB is good.
- And as tape capacity and speeds grow, the minimum desirable file size increases also. “Eventually, any file becomes small.”

# Project Goals

- Automatically aggregate small files into larger “container” files, with configurable definitions of “small” and “larger.”
- Transparently aggregate user's files through existing enstore interface (encp)
- Assume custodial ownership while staged to disk awaiting aggregation
- Preserve end-to-end check-summing
- Per customer "small file" policies

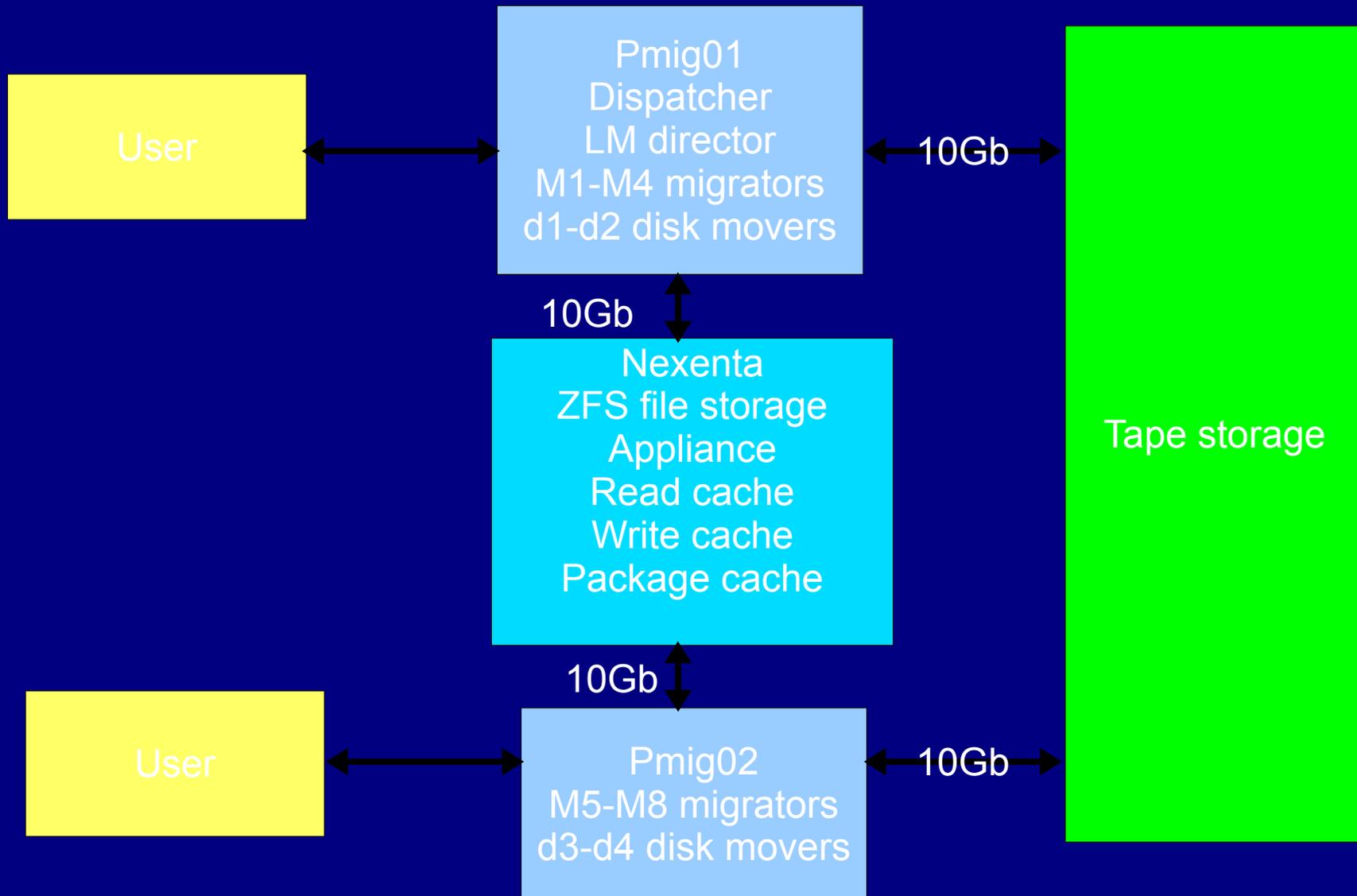
# What files to aggregate ?

- Aggregation of files shall account for read access patterns. Only the experiment, or no one, knows what read patterns will be.
- File aggregation policy must be flexible enough to adapt to different patterns without changing code.
- Aggregation of files by file family and directory trees is good to start with.

# Changes in the project

- All SFA components are currently implemented in python.
- Policy Engine Server and Migration Dispatcher are implemented as 2 threads in one module: dispatcher

# Production configuration



# HUD monitoring page

## Enstore SFA HUD

Volume	KB in transition	# Files in Transition	Used Size(KB)	Total Size(KB)
data_area(write cache)	2682484	27	2491808	22726802592
<b>Volume</b>			<b>Used Size(KB)</b>	<b>Total Size(KB)</b>
archive_area(package staging)			315936	2295418080
<b>Volume</b>	<b># Files</b>	<b>Used Size(KB)</b>	<b>Total Size(KB)</b>	
stage_area(read cache)	3	132416	22726757984	

### FILES IN TRANSITION

# Files in transition

## FILES IN TRANSITION

<a href="#">CD-LTO4F1T.ssa_test.ssa_test.cpio_odc</a>	resulting_library = CD-DiskSF	minimal_file_size = 1953125	max_files_in_pack = 100	rule = {'storage_group': 'ssa_test', 'file_family': 'ssa_test', 'wrapper': 'cpio_odc'}	max_waiting_time = 120
<a href="#">CD-LTO4F1.nova.rawdata_NDOS_unmerged.cpio_odc</a>	resulting_library = CD-DiskSF	minimal_file_size = 4882812	max_files_in_pack = 50	rule = {'storage_group': 'nova', 'file_family': 'rawdata_NDOS_unmerged', 'wrapper': 'cpio_odc'}	max_waiting_time = 86400

### CD-LTO4F1.nova.rawdata\_NDOS\_unmerged.cpio\_odc

**pool=cache\_written, total=7, size=910533 (kB), #of lists=1**

list id=d20b50d1-a1af-4ca3-b763-3611b3f4a974, total=7,size=910533 (kB), time\_qd=2012-04-20 12:42:48

```

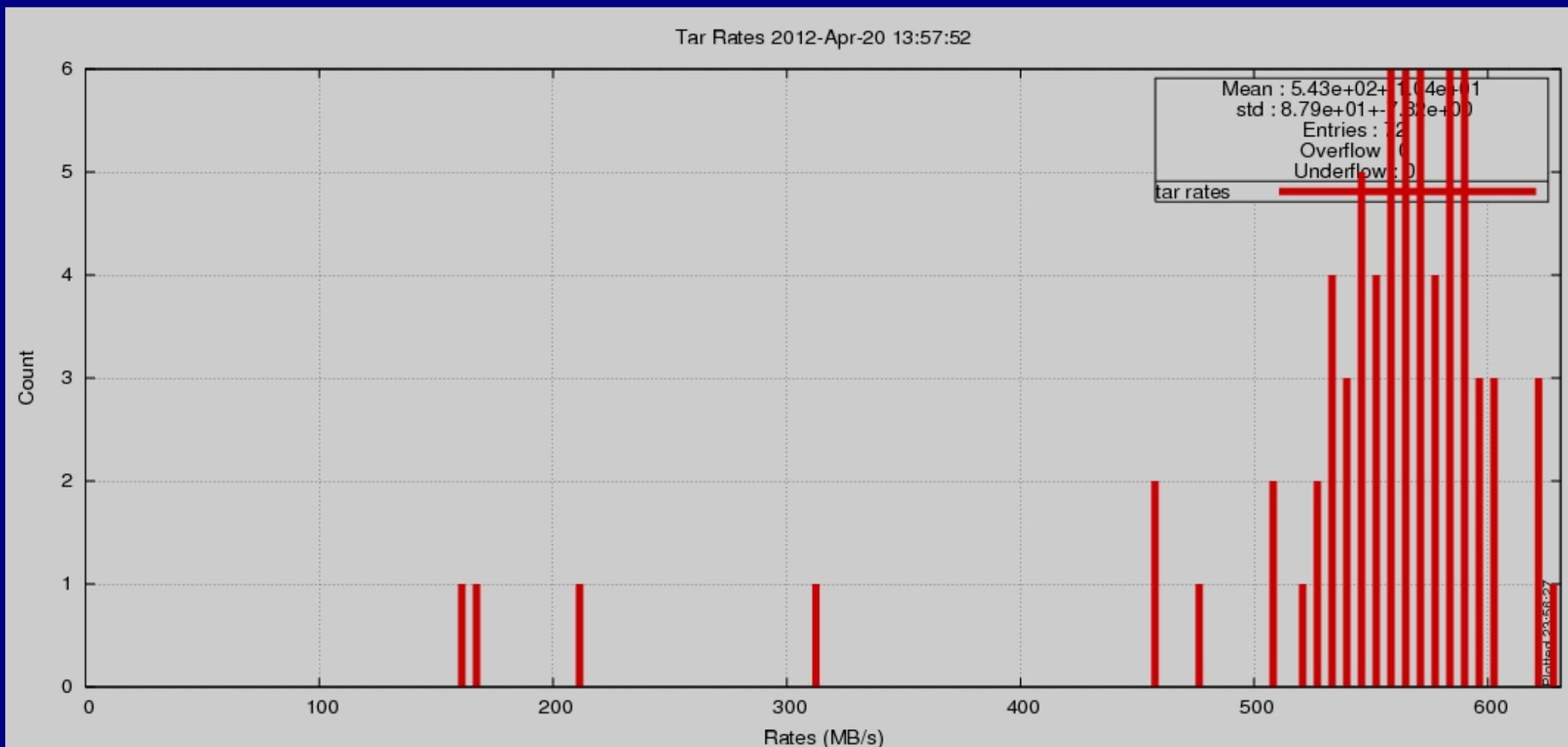
cache_mod time storage_group file family volume location_cookie bfid size crc pnfs_id pnfs_path archive status
2012-04-20 12:42:49 nova rawdata_NDOS_unmerged common:nova.rawdata_NDOS_unmerged.cpio_odc:2012-04-19T18:16:45Z /volumes/aggwrite/cache/16c/e6f/0000C80206895F5244FA985D6E1E93E6FFFF CDM5133494376500000
2012-04-20 12:42:55 nova rawdata_NDOS_unmerged common:nova.rawdata_NDOS_unmerged.cpio_odc:2012-04-19T18:16:45Z /volumes/aggwrite/cache/9a0/bfe/0000CDDCDF211155456DB764EFD285BFE825 CDM5133494377300000
2012-04-20 12:43:02 nova rawdata_NDOS_unmerged common:nova.rawdata_NDOS_unmerged.cpio_odc:2012-04-19T18:16:45Z /volumes/aggwrite/cache/611/3c3/000059217D33DA9D4903A44806F25B3C344A CDM5133494378000001
2012-04-20 12:43:12 nova rawdata_NDOS_unmerged common:nova.rawdata_NDOS_unmerged.cpio_odc:2012-04-19T18:16:45Z /volumes/aggwrite/cache/c27/fd8/0000164F16EBECC74010BB2242C217FD8E30 CDM5133494378800001
2012-04-20 12:43:22 nova rawdata_NDOS_unmerged common:nova.rawdata_NDOS_unmerged.cpio_odc:2012-04-19T18:16:45Z /volumes/aggwrite/cache/94f/1e6/0000222D21D761D74F139C31D648C41E618B CDM5133494379800000
2012-04-20 12:43:35 nova rawdata_NDOS_unmerged common:nova.rawdata_NDOS_unmerged.cpio_odc:2012-04-19T18:16:45Z /volumes/aggwrite/cache/140/4e5/00000353F0F919884861958CC4608A4E51CA CDM5133494381200000
2012-04-20 12:43:28 nova rawdata_NDOS_unmerged common:nova.rawdata_NDOS_unmerged.cpio_odc:2012-04-19T18:16:45Z /volumes/aggwrite/cache/f67/2c5/0000E0F3832513C3487889D86D20592C5F3E CDM5133494380700000

```

# Performance issues

- 10 Gb connection between pmig nodes and appliance but no jumbo frames so far
- Running file aggregation / decomposition on nfs mounted pmig nodes was slow (100 MB/s maximum)
- Run file aggregation / decomposition on appliance (rates >500 MB/s but increased CPU load, see next slide)
- Currently we think we can support a few users.
- We are investigating ways to increase performance and making the system scalable

# Aggregation rates for the first day in production



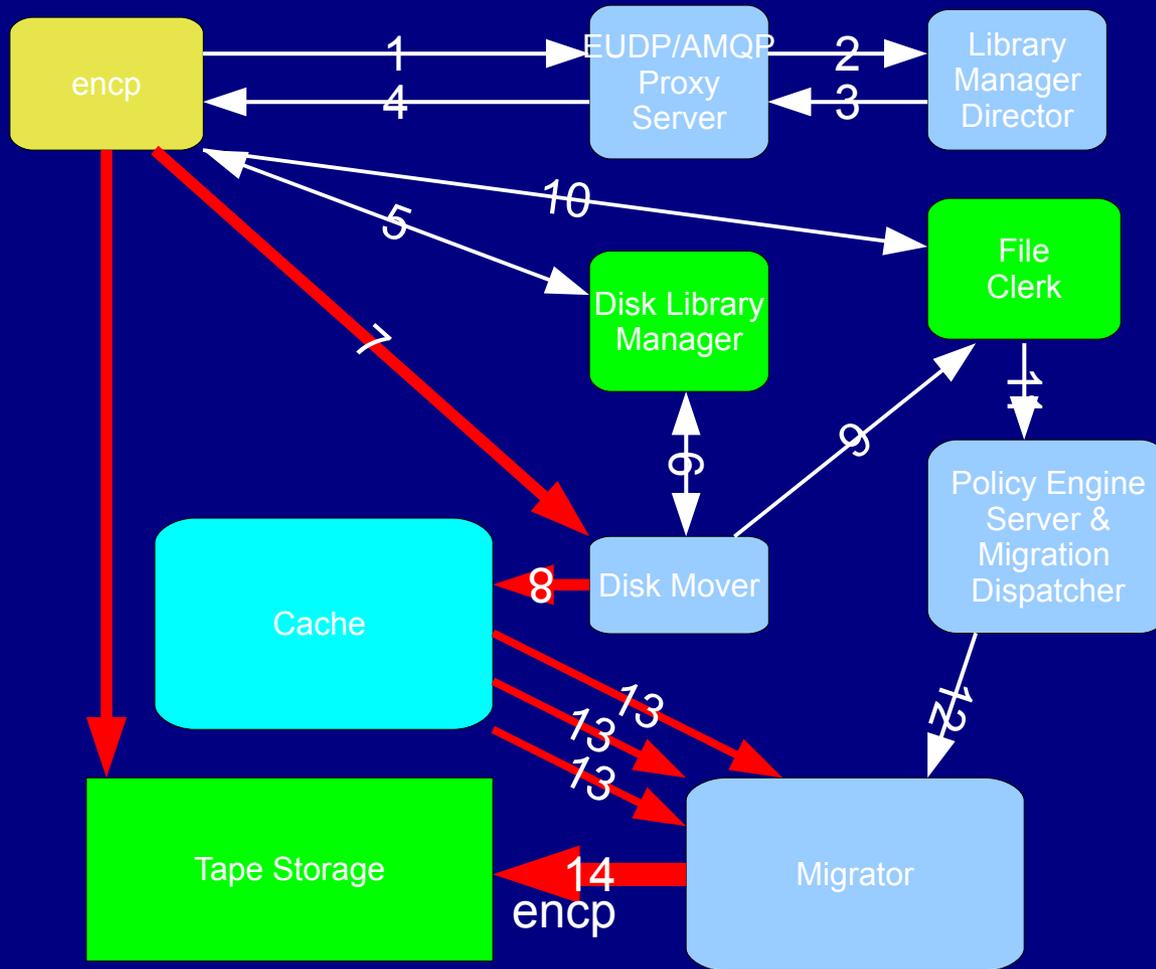
# Further plans

- Any immediate feedback from installation.
- Migration strategy and possible coding.
- Review error handling.
- Improve scalability.
- Investigate how enstore disk mover overhead can be improved.
- End user flush? (We have admin flush).
- Fix duplicate copy CRON to work properly with aggregated files
- Purchase backup/test appliance

# References

- User guide:CS-doc-4698-v3
- File Aggregation Presentation: CS-doc-4596-v1
- Operations Guide: CS-doc-4697-v2
- Review: CS-doc-4652-v1
- Small Files Aggregation Project page

# Writing files.



# Reading files.

